```
=> d his

        (FILE 'USPAT' ENTERED AT 13:40:31 ON 16 DEC 1998)
L1         368 S  (WEB (2A) PAGE#)
L2       16441 S  (AUTOMATIC? OR DYNAMIC?) (2A) (GENERAT? OR CREAT?)
L3          86 S  L1 AND L2
L4          12 S  L1 (P) L2
L5           1 S  L4 (P) REQUEST#
L6           5 S  (5710918 OR 5701451 OR 5572643 OR 5537586 OR 5530852)/PN
L7         185 S  (REDIRECT? OR REROUT?) (2A) REQUEST#
L8           9 S  L1 AND L7
L9          10 S  L7 (P) INTERCEPT?
L10        201 S  (WEB (2A) SERVER#)
L11         98 S  (PAGE (2A) SERVER#)
L12         43 S  L10 AND L11
L13         26 S  L10 (P) L11
L14          0 S  L13 (P) (CONCURRENT? OR SIMULTANEOUS?)
L15          1 S  L4 AND L13
```

US PAT NO: 5,751,956 [IMAGE AVAILABLE]   L15: 1 of 1
TITLE: Method and apparatus for redirection of server external
                 hyper-link references

SUMMARY:

BSUM(21)

 Access counters, however, fundamentally log only **server** local
web page accesses. The client browser to the CGI program is
evaluated by the client in connection with the initial serving of. .   .

DETDESC:

DETD(15)

 Each of the data terms within a redirection URL may be statically or
dynamically created by the HTTPd server 30 as part of the process
of originally serving a **web page** with the embedded redirection
URL to a client computer system 12. With **dynamic generation,**
different graphic images corresponding to a single advertiser or one of
any number of advertisers may be effectively served with an otherwise
statically defined **web page**. The data terms of the embedded
redirection URL may be dynamically selected based on the identity of the
advertiser and. .   . addition to separately establishing a hypertext
link to the graphics image as part of an instance of serving a particular
web page by the HTTPd server 30. Indeed, the selection of
advertiser and graphics image could be made at least in part. .   .

DETDESC:

DETD(16)

 The validation code may also be **dynamically generated**. In an
alternate embodiment of the present invention, the validation code
encodes a representation of the day of the year. .   . to be tracked by
the HTTPd server 30 so as to limit the frequency that a specific
instantiation of the **web page** is repeatedly presented to a
specific client 12. Additionally, the HTTPd server 30 may operate to
block operation on a received redirection URL where the corresponding
web page has not recently been served to the requesting client
12.

TITLE: Method and apparatus for generating dynamic web pages by
                invoking a predefined procedural package stored in a
                database

ABSTRACT:
A method and apparatus for **generating dynamic Web pages**
is disclosed. Specifically, the present invention claims a method and
apparatus for **generating dynamic Web pages** on a **Web**
server by invoking and executing predefined procedural packages stored in
a database. The claimed invention receives an object **request** on the
Web server and activates a Web agent on the Web server based on the
object **request**. The Web agent invokes and executes the predefined
procedural package to retrieve data from a data repository, and then
formats. . .

SUMMARY:

BSUM(8)

 Although **Web pages** were traditionally stored as static files on
the Web server operating system, today **Web pages** can also be
generated dynamically using the Common Gateway Interface (CGI).
CGI is a standard interface for running external programs on a Web
server. It allows Web servers to **create dynamic** documents when
the server receives a **request** from the Web browser. When the Web
server receives a **request** for a dynamic document, the Web server
executes the appropriate CGI script and transmits the output of the
execution back. . . Web browser. The Web browser does not
differentiate between static and dynamic documents. It simply displays
the output of the **request**.

CLAIMS:

CLMS(3)

 3. A computer-implemented method for **generating dynamic Web**
pages on a **Web** server using a predefined procedural package
stored in a database, the computer-implemented method comprising the
steps of:
 receiving a **request** on the Web server;
 activating a Web agent on the Web server based on the **request**, the
  Web agent invoking the predefined procedural package stored in the
  database;
 executing the predefined procedural package to retrieve data from. . .

TITLE: System and method utilizing a virtual addressing buffer
               circuit to emulate a device which is physically not
               present

DETDESC:

DETD(10)

  Referring .   .   . buffer circuit 10 is used as a buffer between the CPU
80 and the memory bus 84 to more efficiently **reroute** CPU address
requests within the host computer system. The control of the virtual
addressing buffer 10 is not fixed, so the operation of.   .   . altered by
the CPU 80 at anytime during the operation of the host computer system.
The virtual addressing buffer 10 **intercepts** a CPU requested address
on the local CPU bus 82. The CPU requested address, also referred to as
the input.   .   .

DETDESC:

DETD(37)

  Thirdly, .   .   . association with the dual comparison virtual
addressing buffer 200. A first portion of the dual comparison virtual
addressing buffer 200 **intercepts** an address request for the address
location of the device which is physically not present ("non-existent
device") and redirects the.   .   . the non-existent device is requested
by the CPU 400, the first portion of the dual comparison virtual
addressing buffer 200 **redirects** the address **request** to a
translated address location and issues the terminate signal at an active
level. The terminate signal is sent to.` .   .

TITLE: System for opening cache file associated with designated
               file of file server only if the file is not subject to
               being modified by different program

ABSTRACT:
A .   .   . from a network server or from the shared cache server. Each
client computer further includes a resident redirector program which
intercepts file manipulation requests from executing application
programs and **redirects** these **requests** to either the shared
network cache or the local non-volatile cache when appropriate.

SUMMARY:

BSUM(18)

  As contemplated by the invention, each file operation request from an
executing application program to the operating system is **intercepted**
by a resident **request redirector** which controls access to the
file server, the shared network cache and the local non-volatile cache to
insure fast, safe.   .   .

DETDESC:

DETD(4)

In . . . serves as ● interface between the applica●n programs and
the operating system fi●e system. The shared LAN cache c●ent module
intercepts file access requests which are issued to the operating
system from executing application programs, **redirecting** these
requests when appropriate to manage the caching of information in
both the shared LAN cache (SLC) server 20 and a local. . .

CLAIMS:

CLMS(11)

11. . . .
persistently storing data,
at least one application program executable on said client computer for
  issuing file open requests and file read **requests,**
a **redirector** program executable on said client computer for
  intercepting said file open requests and said file read
  requests, said **redirector** program including:
  means responsive to a given file open request from said application
    program which specifies a designated file on. . .

US PAT NO: 5,737,769 [IMAGE AVAILABLE]   L9: 3 of 10
TITLE: Physical memory optimization using programmable virtual
                  address buffer circuits to redirect address requests

DETDESC:

DETD(10)

Referring . . . buffer circuit 10 is used as a buffer between the CPU
80 and the memory bus 84 to more efficiently **reroute** CPU address
requests within the host computer system. The control of the virtual
addressing buffer 10 is not fixed, so the operation of. . . altered by
the CPU 80 at anytime during the operation of the host computer system.
The virtual addressing buffer 10 **intercepts** a CPU requested address
on the local CPU bus 82. The CPU requested address, also referred to as
the input. . . .

DETDESC:

DETD(37)

Thirdly, . . . association with the dual comparison virtual
addressing buffer 200. A first portion of the dual comparison virtual
addressing buffer 200 **intercepts** an address request for the address
location of the device which is physically not present ("non-existent
device") and redirects the. . . the non-existent device is requested
by the CPU 400, the first portion of the dual comparison virtual
addressing buffer 200 **redirects** the address **request** to a
translated address location and issues the terminate signal at an active
level. The terminate signal is sent to. . .

CLAIMS:

CLMS(1)

What . . .
programmable buffer address circuit having at least a first writable
  memory element configured to store first address mapping data, to
  intercept address requests from said processing unit for the
  reserved memory mapped space and to changeably **redirect** said
  address **requests** according to said first address mapping data from
  a local memory bus to a slower downstream bus where a requested. . .
  buffer address circuit having at least a second writable memory element

configured to store second address mapping data, to changeably **redirect** address req●●s for memory locations above ●physical memory space according to said second address●apping data to the physical memory space.   .   .

CLAIMS:

CLMS(2)

2. .   .   .
at least a first writable memory element configured to store first address mapping data, said first virtual address buffer circuit **intercepting** address requests by said CPU for said second memory space, said first virtual address buffer circuit **redirecting** said **requests** from said local bus to said peripheral located on said downstream bus according to said first address mapping data; and .
a. .   .   having at least a second writable memory element configured to store second address mapping data, said second virtual address buffer **intercepting** address requests from said CPU to said first memory space and said second virtual address buffer **redirecting** said **requests** to said second memory space according to said second address mapping data.

US PAT NO: 5,680,303 [IMAGE AVAILABLE]   L9: 4 of 10
TITLE: Communication device sharing on a local area network

DETDESC:

DETD(10)·

For .   .   .   the I/O manager 16, 28 processes FO requests which may include requests directed to the X.25 card 36. The NT **redirector** 20 **intercepts request** for non-local (i.e. shared devices). It then "redirects" them to the machine on which the device actually resides. The Network.   .   .

US PAT NO: 5,671,345 [IMAGE AVAILABLE]   L9: 5 of 10
TITLE: System and method for intercepting and reconstructing
                graphics management tool marking instructions

SUMMARY:

BSUM(12)

The glue code of the present invention **intercepts** the marking requests sent by the interpreter software and acknowledges receipt of the marking requests to make it appear to.   .   .   being written into the frame buffer. In addition, the glue code accumulates state history for each request and interprets the **redirected** marking **requests** using the accumulated state history so as to reconstruct it into a high-level object oriented display list having a second.   .   .

US PAT NO: 5,642,417 [IMAGE AVAILABLE]   L9: 6 of 10
TITLE: Virtualized installation of material

DETDESC:

DETD(43)

A .   .   .   Pu. F5. Any requests for information unaffected by a standard installation are passed onto the user resource Mu.a, otherwise the **request** is **redirected** via a task r5 to a redirection module Mr (28) that includes the redirection resource Mr.a. The task R6 monitors. .   .   Any requests for information that are unaffected by a standard installation are passed onto the user resource Mu.b. Otherwise, the

request is **redirected** via a task r6 to a redirection resource
Mr.b. The task R7 **inte**⬤**pts** all user requests for reso⬤es that are
normally added by a standard installation, and **redirects**⬤he
requests via a task r7 to a redirection resource Mr.c. Note that the
user task F8 is not monitored because it.   .   .

US PAT NO: 5,603,014 [IMAGE AVAILABLE]   L9: 7 of 10
TITLE: Protected mode simulation of a real mode interupt based
                 programming interface in a computer system

DETDESC:

DETD(29)

 The communication trap driver 44 **intercepts** the real mode software
interrupt 2F generated by the communication application 40. The
communication trap driver 44 **redirects** the function **requests** of
the real mode software interrupt 2F to the HOST-OS communication resident
manager 42 in the protected mode of the.   .   .

US PAT NO: 5,546,398 [IMAGE AVAILABLE]   L9: 8 of 10
TITLE: Signal intercept system and method

SUMMARY:

BSUM(12)

 In .   .   .   desired operation; initializing two separate buffers and
direct memory accesses for transmit channels and receive channels;
opening the channels for **intercepting** the signals and receive the
receive frames to receive channel; transmitting the first frame with a
desired delay; testing for.   .   .   is a valid card; if so, make no
changes. If the current called number is found in the exception table,
reroute the **request** to a non-existent number or redirecting the
related call to an operator; making the main processor periodically
process the calling.   .   .

US PAT NO: 5,526,503 [IMAGE AVAILABLE]   L9: 9 of 10
TITLE: Virtual addressing buffer circuit

DETDESC:

DETD(11)

 Referring .   .   .   buffer circuit 10 is used as a buffer between the CPU
80 and the memory bus 84 to more efficiently **reroute** CPU address
requests within the host computer system. The control of the virtual
addressing buffer 10 is not fixed, so the operation of.   .   .   altered by
the CPU 80 at anytime during the operation of the host computer system.
The virtual addressing buffer 10 **intercepts** a CPU requested address
on the local CPU bus 82. The CPU requested address, also referred to as
the input.   .   .

DETDESC:

DETD(38)

 Thirdly, .   .   .   association with the dual comparison virtual
addressing buffer 200. A first portion of the dual comparison virtual
addressing buffer 200 **intercepts** an address request for the address
location of the device which is physically not present ("non-existent
device") and redirects the.   .   .   the non-existent device is requested
by the CPU 400, the first portion of the dual comparison virtual
addressing buffer 200 **redirects** the address **request** to a
translated address location and issues the terminate signal at an active

level. The terminate signal is sent to.   .   .

TITLE: Method for executing programs within expanded memory of a
                 computer system using MS or PC DOS

DETDESC:

DETD(29)

 FIG. 8 depicts a flowchart of the module that is loaded in DOS to
intercept I/O requests having a data address in expanded memory. The
module of FIG. 8 is linked into the DOS I/O chain so that it receives
control each time any I/O request is made, allowing it to examine all I/O
requests and **redirect** their data buffer addresses when such
addresses reference expanded memory.

US PAT NO: **5,710,918** [IMAGE AVAILABLE]   L6: 1 of 5
TITLE: Method for distributed task fulfillment of web browser
               requests

US PAT NO: **5,701,451** [IMAGE AVAILABLE]   L6: 2 of 5
TITLE: Method for fulfilling requests of a web browser

US PAT NO: **5,572,643** [IMAGE AVAILABLE]   L6: 3 of 5
TITLE: Web browser with dynamic display of information objects
               during linking

US PAT NO: **5,537,586** [IMAGE AVAILABLE]   L6: 4 of 5
TITLE: Enhanced apparatus and methods for retrieving and
               selecting profiled textural information records from a
               database of defined category structures

US PAT NO: **5,530,852** [IMAGE AVAILABLE]   L6: 5 of 5
TITLE: Method for extracting profiles and topics from a first
               file written in a first markup language and generating
               files in different markup languages containing the
               profiles and topics for use in accessing data described
               by the profiles and topics

US PAT NO: 5,835,712 [IMAGE AVAILABLE]   L8: 2 of 9
TITLE: Client-server system using embedded hypertext tags for
                application and database development

SUMMARY:

BSUM(7)

The .   .   .   the WWW are called the Web sites. The electronic documents
provided by the Web sites are commonly referred to as **Web pages**
or files. A client software which navigates through the Internet sites
and displays **Web pages** is referred to as the Web "browser." A
browser allows access not only to **Web pages**, but all the other
existing information resources on the Internet.

SUMMARY:

BSUM(8)

The .   .   .   with a special set of codes which indicate how the document
should be displayed. Upon receiving a request for a **Web page**, a
server typically returns an HTML document which is decoded and displayed
on a Web browser running on the client's.   .   .

SUMMARY:

BSUM(9)

One .   .   .   document or to another document by selecting a link and
causing the browser to transmit a request for a new **Web page**
through the associated URL. In the WWW environment, the HTML documents
very often add multimedia elements, such as graphics, sound,.   .   .

SUMMARY:

BSUM(10)

While .   .   .   depending on the information, the content providers often
find it necessary to incorporate and merge data from multiple sources
into **Web pages**, further adding to the updating chores.

SUMMARY:

BSUM(11)

Additionally, .   .   .   HTTP is a stateless, object-oriented protocol in
which much of the Web transactions involve transferring a series of
static HTML **pages**. When a **Web** server returns a requested **Web
page** to the client, the link between the client and server is no
longer maintained. The client may of course choose.   .   .   re-establish a
link. As a result, however, a critical limitation of the WWW is that the
information contained in a **Web page**, regardless of how relevant
it is to the pages following, cannot be maintained from page to page
within a WWW.   .   .

SUMMARY:

BSUM(13)

The . . . provide ●ynamic client/server environme● without the
complexities associated with CGI programming, and signi●●antly removes
the laborious task of updating **Web pages** on the WWW. In a
preferred embodiment, the present invention provides a framework for
rapidly deploying new applications based on. . .

SUMMARY:

BSUM(14)

In . . . each tag with the corresponding value. In the preferred
embodiment, a Web server, in response to a request for the **Web
page** from a client, processes such a source by executing the tag
extensions to expand.

SUMMARY:

BSUM(16)

In . . . are collections of data for use in a particular application.
As mentioned, a tag extension in a source to a **Web page** is
associated with a value in a database. Such a value can be static data or
a variable, such as. . .

SUMMARY:

BSUM(19)

The . . . database. Both the template and content databases are
controlled by the server of the present invention. Preferably, sources to
all **Web pages** controlled by the processor of the present
invention are constructed using templates. As previously mentioned, a
source contains HTML tags. . .

SUMMARY:

BSUM(22)

Another . . . protected record in a database where the record
includes an access control list to specify authorized user
identifications. Such a **request** is **redirected** to a verification
directory which causes the server to issue an input query to the client
to input a user. . .

US PAT NO: 5,838,916 [IMAGE AVAILABLE]   L8: 1 of 9
TITLE: Systems and methods for executing application programs
                from a memory device linked to a server

ABSTRACT:
Systems .   .   . a local program memory element, a file block associated
with the selected remote file pointer. The process can employ a
redirector that translates **requests** to access remote files into
HTTP compliant commands for collecting files from an HTTP server site.

SUMMARY:

BSUM(18)

 The .   .   . operating system to detect file system requests for files
stored within the remote file system and passing the file system
requests to a **redirector** element for translating the file system
requests into HTTP compliant signals for transmission across a network.
In this practice of.   .   .

DETDESC:

DETD(2)

 The .   .   . comprises, inter alia, systems and methods that enable a
web site administrator to provide links to remote applications within
their **web pages**. One realization of the invention is that a
remote client can be allowed to execute an application program stored at.
.   .

DETDESC:

DETD(9)

 FIG. .   .   . the systems of the invention extend the capabilities of
the World Wide Web by enabling web site administrators to provide **web
pages** that have links to remote applications. Moreover, the web site
administrator can employ the systems of the invention to provide.   .   .

DETDESC:

DETD(11)

 FIG. 2 depicts one process for providing a **web page** application
link. In this process, the application program is provided within a
shared directory, or an exported file system. The.   .   .

DETDESC:

DETD(25)

 Accordingly .   .   . an administrator, or optionally a client, to create
and configure application information files which can be specified as
links in **web pages**. In a further embodiment, the administration
program 44 can contain a process that monitors system use and generates
statistics that.   .   .

DETDESC:

DETD(48)

    FIG. 7 depicts, that in addition to running remote applications from the network through **web pages** presented by a server 214, applications can also be run from the cache memory 216, optionally by activating links in. . .

DETDESC:

DETD(57)

    To . . . or abstractions, and sends the request to a server. As shown, each redirector in the network file system 250 can **redirect** I/O **requests** for use with a selected type of protocol for sharing files, data or devices.

US PAT NO: 5,838,916 [IMAGE AVAILABLE]   L8: 1 of 9
TITLE: Systems and methods for executing application programs
              from a memory device linked to a server

US PAT NO: 5,835,712 [IMAGE AVAILABLE]   L8: 2 of 9
TITLE: Client-server system using embedded hypertext tags for
              application and database development

US PAT NO: 5,802,299 [IMAGE AVAILABLE]   L8: 3 of 9
TITLE: Interactive system for authoring hypertext document
              collections

US PAT NO: 5,781,909 [IMAGE AVAILABLE]   L8: 4 of 9
TITLE: Supervised satellite kiosk management system with combined
              local and remote data storage

US PAT NO: 5,778,368 [IMAGE AVAILABLE]   L8: 5 of 9.
TITLE: Real-time embedded software respository with attribute
              searching apparatus and method

US PAT NO: 5,761,683 [IMAGE AVAILABLE]   L8: 6 of 9
TITLE: Techniques for changing the behavior of a link in a
              hypertext document

US PAT NO: 5,751,956 [IMAGE AVAILABLE]   L8: 7 of 9
TITLE: Method and apparatus for redirection of server external
              hyper-link references

US PAT NO: 5,740,430 [IMAGE AVAILABLE]   L8: 8 of 9
TITLE: Method and apparatus for server-independent caching of
              dynamically-generated customized pages

US PAT NO: 5,708,780 [IMAGE AVAILABLE]   L8: 9 of 9
TITLE: Internet server access control and monitoring systems